

## CLAIM AMENDMENTS

### Claim Amendment Summary

#### Claims pending

- At time of the Action: Claims 1-15 and 18-22.
- After this Response: Claims 1-4, 6-15 and 18-22.

Canceled or Withdrawn claims: 5.

Amended claims: 1, 18, and 21.

New claims: none.

---

### CLAIMS:

1. (CURRENTLY AMENDED) A method for generating a delta between a first program binary and a second program binary, the method comprising the steps of:

obtaining a first control flow graph (CFG) representation of the first binary and obtaining a second CFG representation of the second binary;

comparing the first and second CFG representations to identify blocks (nominally matched blocks) that match in the first and second CFG representations, thereby identifying blocks (nominally unmatched blocks) in the second CFG representation that do not match in the first CFG representation, the comparing being based upon content of blocks being compared and augmented local neighborhoods of blocks surrounding blocks being compared, wherein a local neighborhood of a particular block consists of blocks neighboring that block

421 West Riverside, Suite 500  
Spokane, WA 99201  
P: 509.324-9256  
F: 509.323-8979  
www.leeandhayes.com

**lee & hayes**  
ATTORNEYS AT LAW

1 in a CFG representation, but less than all the blocks in that CFG representation,  
2 and an augmented local neighborhood of that particular block consists that block's  
3 local neighborhood plus a random sampling of blocks from a substantially larger  
4 neighborhood of blocks surrounding that block, an augmented local neighborhood  
5 in a CFG representation consisting of less than all the blocks in that CFG  
6 representation;

7 determining edit-operations that merges the unmatched blocks into the first  
8 CFG representation so that first CFG representation is substantially identical to the  
9 second CFG representation;

10 producing a delta comprising the unmatched blocks and the edit-operations.

11  
12 2. (ORIGINAL) A method for transmitting a delta comprising:

13 a method for generating a delta as recited in claim 1;

14 transmitting the delta over a network.

15  
16 3. (ORIGINAL) A method for patching a copy of the first program  
17 binary, the method comprising:

18 a method for generating a delta as recited in claim 1;

19 patching the copy of the first program binary so that the copy is  
20 substantially identical to the second program binary, wherein the delta guides such  
21 patching.

421 West Riverside, Suite 500  
Spokane, WA 99201  
P: 509.324-9256  
F: 509.323-8979  
www.leeandhayes.com  
lee & hayes

1  
2 4. (PREVIOUSLY PRESENTED) A method as recited in claim 1,  
3 wherein a local neighborhood of a particular block consists of those blocks  
4 immediately adjacent that block.

5  
6 5. (CANCELED)

7  
8 6. (ORIGINAL) A computer-readable medium having embodied  
9 thereon a data structure, comprising a delta generated in accordance with the steps  
10 recited in claim 1.

11  
12 7. (ORIGINAL) A computer-readable medium having computer-  
13 executable instructions that, when executed by a computer, performs the method  
14 as recited in claim 1.

15  
16 8. (ORIGINAL) A method for matching blocks between a first control  
17 flow graph (CFG) representation of a portion of a first program and a second CFG  
18 representation of a portion of a second program, the method comprising:

19 matching blocks between the first and second CFG representations based  
20 upon the content of the blocks;

21 detecting outliers, wherein outliers are blocks in the first CFG  
22 representation that do not match any block in the second CFG representation  
23 during the matching step;

24 computing a neighborhood of each block in the first and second CFG  
25 representations by performing a breadth first traversal;

removing the outliers from each neighborhood.

9. (PREVIOUSLY PRESENTED) A method as recited in claim 8 further comprising:

computing labels for each block in first and second CFG representations based upon content of a block;

for each neighborhood computed in the computing step, forming a "d-label" for each block in a neighborhood based upon labels of the blocks within the neighborhood;

attempting to match blocks between first and second CFG representations by comparing the d-labels of the blocks.

10. (ORIGINAL) A computer-readable medium having computer-executable instructions that, when executed by a computer, performs the method as recited in claim 8.

11. (PREVIOUSLY PRESENTED) A method for matching procedures between a first control flow graph (CFG) representation of a portion of a first program and a second CFG representation of a portion of a second program, wherein a procedure comprises multiple blocks in a CFG representation, the method comprising:

computing a procedure-match-criterion for each procedure in the second CFG representation, where the procedure-match-criterion for a procedure in the second CFG representation represents the number of matching blocks between that procedure and a specified procedure in the first CFG representation;

421 West Riverside, Suite 500  
Spokane, WA 99201  
P: 509.324-9256  
F: 509.323-8979  
www.leeandhayes.com

**lee & hayes**  
ATTORNEYS AT LAW

Serial No.: 09/713,633  
Atty Docket No.: MS1-650US  
RESPONSE TO FINAL OFFICE ACTION DATED  
11/2/2004

5

1210041212 G:\MS1-01650\uslms1-650us.m03.doc

atty: Kasey C. Christie

1 matching procedures in the second CFG representation with the specified  
2 procedure in the first CFG representation based upon the procedure-match-criteria  
3 for the procedures in the second CFG representation.

4  
5 12. (ORIGINAL) A computer-readable medium having computer-  
6 executable instructions that, when executed by a computer, performs the method  
7 as recited in claim 11.

8  
9 13. (PREVIOUSLY PRESENTED) A method for matching of blocks  
10 in a procedure of a first control flow graph (CFG) representation of a portion of a  
11 first program between an ostensibly matching procedure of a second CFG  
12 representation of a portion of a second program, the method comprising:

13 matching blocks between the first and second CFG representations based  
14 upon the content of the blocks;

15 computing successively smaller neighborhoods of each block in the first  
16 and second CFG representations via breadth first traversals;

17 for each neighborhood computed in the computing step, forming a "d-  
18 label" for each block in a neighborhood based upon labels of the blocks within the  
19 neighborhood;

20 attempting to match blocks between first and second CFG representations  
21 by comparing the d-labels of the blocks.

22  
23  
24  
25  
Serial No.: 09/713,633

Atty Docket No.: MS1-650US

RESPONSE TO FINAL OFFICE ACTION DATED

6

1210041212 G:\MS1-0650US\ms1-650US.MOS.doc

atty: Kasey C. Christo

1           **14. (ORIGINAL)** A method as recited in claim 13, wherein at least one  
2 neighborhood computed in the computing steps is augmented with a random  
3 sampling of blocks in the complete representation of the neighborhood.

4  
5           **15. (ORIGINAL)** A computer-readable medium having computer-  
6 executable instructions that, when executed by a computer, performs the method  
7 as recited in claim 13.

8  
9           **16. (NOT ENTERED)**

10  
11           **17. (NOT ENTERED)**

12  
13           **18. (CURRENTLY AMENDED)** A patch data structure generated in  
14 accordance with the following acts:

15           providing a server computer in a communications with a communications  
16 network;

17           receiving input from a client computer by way of the communications  
18 network, the input providing a parameter indicative of a request for upgrading a  
19 copy of a first program binary to a match a second program binary;

20           retrieving a delta between a first program binary and the second program  
21 binary, wherein computing such delta comprises the steps of:

- 22           a) obtaining a first control flow graph (CFG) representation of the first  
23 binary and obtaining a second CFG representation of the second  
24 binary;

1 b) comparing the first and second CFG representations to identify  
2 blocks (nominally matched blocks) that match in the first and second  
3 CFG representations, thereby identifying blocks (nominally  
4 unmatched blocks) in the second CFG representation that do not  
5 match in the first CFG representation, the comparing being based  
6 upon content of blocks being compared and augmented local  
7 neighborhoods of blocks surrounding blocks being compared,  
8 wherein a local neighborhood of a particular block consists of blocks  
9 neighboring that block in a CFG representation, but less than all the  
10 blocks in that CFG representation, and an augmented local  
11 neighborhood of that particular block consists that block's local  
12 neighborhood plus a random sampling of blocks from a substantially  
13 larger neighborhood of blocks surrounding that block, an augmented  
14 local neighborhood in a CFG representation consisting of less than  
15 all the blocks in that CFG representation;

16 c) determining edit-operations that merges the unmatched blocks into  
17 the first CFG representation so that first CFG representation is  
18 substantially identical to the second CFG representation;

19 d) producing a delta comprising the unmatched blocks and the edit-  
20 operations;

21 generating the patch data structure as a function of the delta.  
22  
23  
24  
25

1           **19. (ORIGINAL)** A method for transmitting a patch data structure  
2 comprising transmitting a patch data structure as recited in claim 18 over a  
3 communications network.

4  
5           **20. (ORIGINAL)** A method for patching a copy of the first program  
6 binary at a client computer, the method comprising patching the copy of the first  
7 program binary so that the copy is substantially identical to the second program  
8 binary, wherein a delta in a patch data structure as recited in claim 18 guides such  
9 patching.

10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

421 West Riverside, Suite 500  
Spokane, WA 99201  
P: 509.324-9258  
F: 509.323-8979  
www.lee&hayes.com

**lee & hayes**



1           **21. (CURRENTLY AMENDED)** A delta-generator system, comprising:

2           a comparator that is configured to compare a first control flow graph (CFG)  
3 representation of a first program binary and a second CFG representation of the  
4 second program binary for identifying blocks (nominally matched blocks) that  
5 match in the first and second CFG representations, thereby identifying blocks  
6 (nominally unmatched blocks) in the second CFG representation that do not match  
7 in the first CFG representation, the comparing being based upon content of blocks  
8 being compared and augmented local neighborhoods of blocks surrounding blocks  
9 being compared, wherein a local neighborhood of a particular block consists of  
10 blocks neighboring that block in a CFG representation, but less than all the blocks  
11 in that CFG representation, and an augmented local neighborhood of that  
12 particular block consists that block's local neighborhood plus a random sampling  
13 of blocks from a substantially larger neighborhood of blocks surrounding that  
14 block, an augmented local neighborhood in a CFG representation consisting of  
15 less than all the blocks in that CFG representation;

16           an edit-op determiner configured to determine the edit-operations that  
17 merges the unmatched blocks into the first CFG representation so that first CFG  
18 representation is substantially identical to the second CFG representation;

19           an output sub-system that is configured to produce a delta comprising the  
20 unmatched blocks and the edit-operations.

21  
22           **22. (ORIGINAL)** A computer-readable medium having embodied  
23 thereon a data structure comprising a delta produced by the system as recited in  
24 claim 21.  
25

421 West Riverside, Suite 500  
Spokane, WA 99201  
P: 509.324-9256  
F: 509.323-8979  
www.lee&hayes.com  
**lee&hayes**